How to Choose a Threshold for the LLM Evaluation Metrics

Dhagash Mehta
BlackRock, Inc.

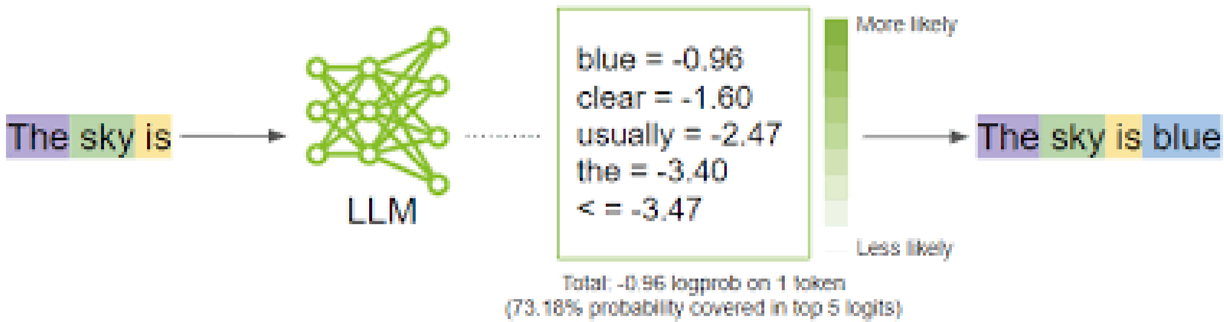
In Collaboration with: Bhaskarjit Sarmah, Mingshu Li, Jingrao Lyu, Sebastian Frank, Nathalia Castellanos, Stefano Pasquali

https://arxiv.org/abs/2412.12148

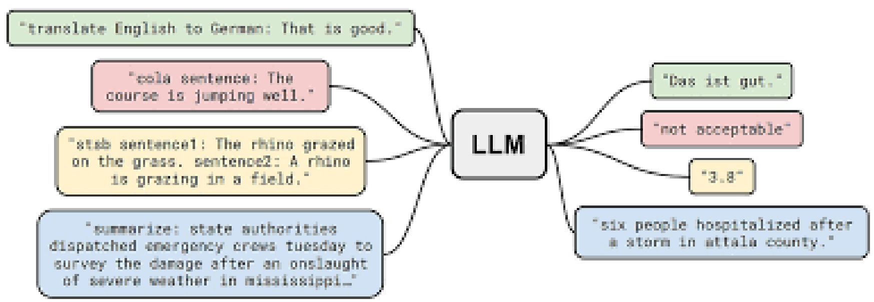
Disclaimer: All the views in this presentation are those of the presenter and not of Blackrock, Inc.

Large Language Models (LLMs)

- An LLM is a GenAI algorithm which though trained in a supervised fashion (next word prediction), it learns the joint distribution of the given large corpus of text data.



- Once trained, it can be used for many different down-stream tasks not just next word prediction.



- Usually, an LLM is trained with massive computational efforts. Most non-tech companies may not have resources to train LLMs from scratch (…yet!).

- Hence, most companies rely on third-party pre-trained, i.e., foundation models such as Llama-3 (Meta), GPTs (OpenAI), Gemini (Google), Claude (Anthropic), etc.

Retrieval Augmented Generation: where can things go wrong

Input Query/Prompt

- Toxicity
- Prompt Injection
- Personal Identifier Information
- Prompt Injection/Jailbreak/Adversarial attacks
- Off-topic (e.g., medical advice)
- Domain specific legally or otherwise prohibited queries (e.g., investment advice)

Output/Answer

- Toxicity
- Personal Identifier Information
- Copyrighted information
- Incorrect/irrelevant answers
- Domain specific legally or otherwise prohibited queries (e.g., investment advice)
- Bias/Fairness (e.g., information about muni assets having bias for ‘Blue’ vs ‘Red’ states)
- Explanability
- Uncertainty quantification

Retrieval Augmented Generation: Objective Evaluation Metrics

- There are numerous evaluation metrics for the LLM systems (e.g., RAG or text summarization, etc.) proposed in the literature.
- They can be broadly classified into two categories:

1. 'Offline' evaluation metrics: they require ground truth QA pairs and/or context.

- These evaluation metrics are useful to validate an LLM system in an 'off-line mode'.

- e.g., answer similarity metrics (Euclidean distance, longest common sequence, Bleu, Rouge, BERTScore, etc.)

2. 'Online' evaluation metrics: they do not require ground truth Q&A pairs or context.

- These evaluation metrics are useful to continuously monitor the LLM application system when it is online and we cannot have ground truths any more.

- e.g., Groundedness/Faithfulness, Diversity, Coherence, etc.

A Concrete Example: Groundedness/Faithfulness

- **<u>Is the answer supported by the context?</u>**

- For a given query, q, the answer, as(q), is <u>faithful</u> to the context, c(q), if the claims that are made in the answer can be inferred from the context.

- To estimate faithfulness, we first use an LLM to extract a set of statements, i.e., decompose longer sentences in the answer into shorter and more focused assertions.

- For each statement, s_i , the LLM determines if s_i can be inferred from c(q). Prompt (RAGAS):

```
Given a question and answer, create one or more statements from each sentence
in the given answer.

    Question: "{question}"
    Answer: "{answer}"

    Consider the given context and following statements, then determine whether
they are supported by the information present in the context. Provide a brief
explanation for each statement before arriving at the verdict (Yes/No). Provide
a final verdict for each statement in order at the end in the given format. Do
not deviate from the specified format.

    Context: "{context}“
```


- Then, Faithfulness = (No. of verified sentences)/(Total no. of sentences in the answer)

Retrieval Augmented Generation: Groundedness/Faithfulness

- **Is the answer supported by the context?**

E.g.,
question = "What is the capital of France?“
context = "The capital of France is Paris. Paris is known for its culture, history, and landmarks such as the Eiffel Tower."
answer = "The capital of France is Paris. It is a large city with a significant cultural heritage."

- Generated statements from the :
  1. Paris is the capital of France.
  2. Paris is a city with a rich cultural heritage.
  3. The Eiffel Tower is a landmark in Paris.

- Explanations:
  1. The first statement is directly supported by the context, which states that "The capital of France is Paris.“. Verdict: Yes
  2. The second statement is indirectly supported by the context. While the context does not explicitly state that Paris has a "rich cultural heritage," it does mention that Paris is known for its culture and history, which can be interpreted as a rich cultural heritage. Verdict: Yes
  3. The third statement is directly supported by the context, which mentions the Eiffel Tower as a landmark in Paris. Verdict: Yes

Retrieval Augmented Generation: Groundedness/Faithfulness

- **Is the answer supported by the context?**

E.g.,
question = "What is the capital of France?“
context = "The capital of France is Paris. Paris is known for its culture, history, and landmarks such as the Eiffel Tower."
answer = "The capital of France is Paris. It is a large city with a significant cultural heritage."

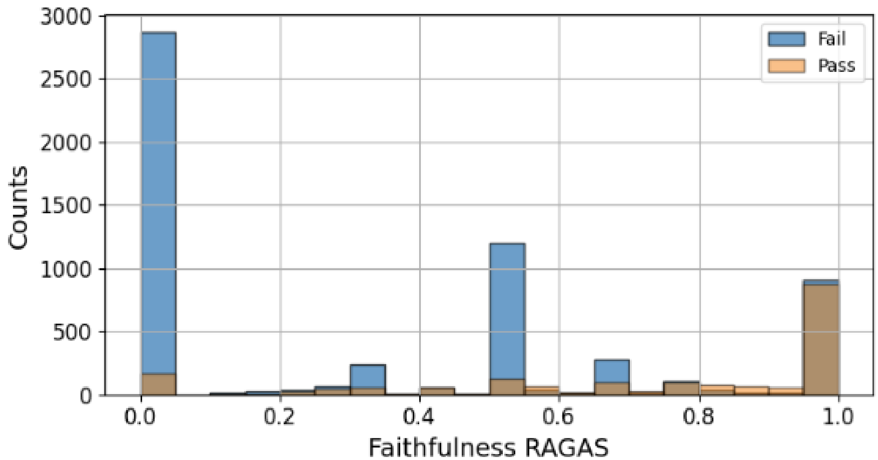
- Generated statements from the :
  1. Paris is the capital of France.
  2. Paris is a city with a rich cultural heritage.
  3. The Eiffel Tower is a landmark in Paris.

Faithfulness = 3/3 = 1.0

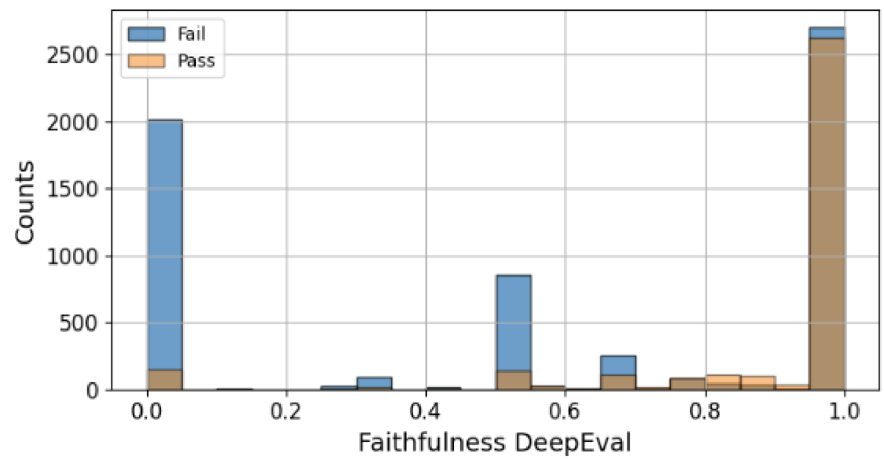
- Explanations:
  1. The first statement is directly supported by the context, which states that "The capital of France is Paris.“. Verdict: Yes
  2. The second statement is indirectly supported by the context. While the context does not explicitly state that Paris has a "rich cultural heritage," it does mention that Paris is known for its culture and history, which can be interpreted as a rich cultural heritage. Verdict: Yes
  3. The third statement is directly supported by the context, which mentions the Eiffel Tower as a landmark in Paris. Verdict: Yes

Retrieval Augmented Generation: How to set thresholds …

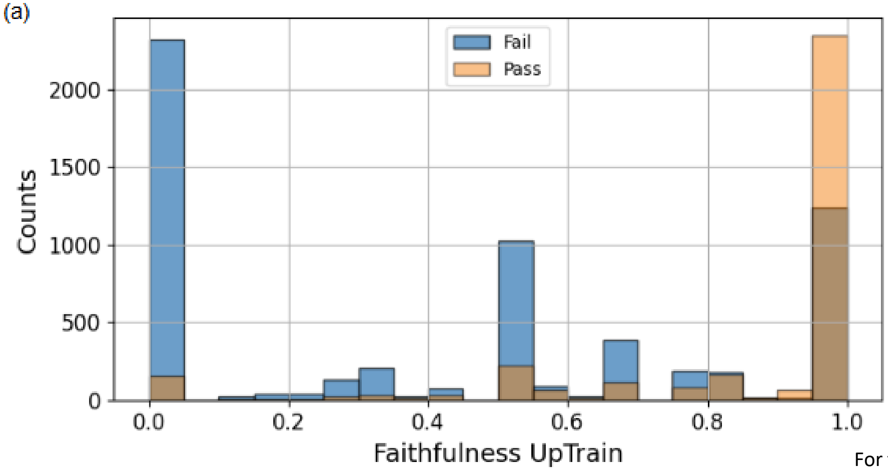
- Dataset: HaluBench is a publicly available dataset with 15k samples each with Context-Question-Answer triplets, and human annotation Hallucinated (Fail)/Not Hallucinated (PASS).
- Run the RAGAS, DeepEval and UpTrain Faithfulness computations, for example, and we get the following distribution (for 9616 samples – after data cleaning).



(a)



(b)



(c)

Retrieval Augmented Generation: How to identify thresholds …

Step-1A: Identify risks of the specific application and a specific evaluation metric that can quantify the risks.

- There may be multiple risks for the business unit for a given AI application; Legal/compliance/regulatory, reputational, financial, etc.;

- E.g., The chatbot might generate answers not supported by the retrieved documents;
  - Potential dissemination of outdated or incorrect financial data.

- Essentially, prescribe a methodology to assign a risk rating for each AI application.

- Then, identify specific evaluation metric(s) to measure the attributes to quantify the risks. E.g., for hallucination related risks, a possible evaluation metric may be Faithfulness.

Step-1B: Identify risk tolerance of the stakeholder(s)

- Identify the risk tolerance of the stakeholder(s) for the specific application by using methods potentially inspired by methods to identify financial risk tolerance.

- **PS:** An academically sound and rigorous way to identify the risk tolerance of the individuals is well-discussed in the Prospect Theory in the Behavioral Economics areas (Kahneman and Tversky, 1979). It can be extended to identify risk tolerance towards AI applications, but research is still underway.

- Say, High/Moderate/Low Risk Appetite.

- Also find trade-off between LLM evaluation cost vs risk.

Step-1C: Map the risk tolerance of the stakeholder(s) to a confidence level.

- The final goal of this exercises is to provide an answer to the following question:

  What percentage of Type I (false positive) and Type II(false negative) errors the stakeholder is willing to take for the specific application with respect to the chosen metric?

- To feed a concrete statistic into the downstream computation, we need a statistical confidence level (e.g., only 5% hallucination is accepted for a specific application for moderate risk tolerance, and hence the required confidence level is 95%).

Retrieval Augmented Generation: Generate Ground Truths

Step-2: Prepare Ground Truth Dataset

- It is crucial for any RAG systems to have some ground truth pairs (Question-Answer) or, even better, triplets (Question-Context-Answer (QCA)) to calibrate the system.

- There are various ways to create such ground truth datasets:
  - Manual creation and labeling
  - Synthetic (using another LLM) generation of QCA and manual labeling

- Generate QCA from <u>diverse set of documents</u> (e.g., out of 10K available documents, randomly pick a few 100s and generate QCAs);

- Generate <u>diverse types questions</u> (around 50 different types of questions classified in the computational linguistics literature, e.g., abbreviation, entity, description, human, location, numeric, etc.);

- Generate questions from <u>different possible topics</u> from the documents (e.g., first perform topic modeling on all the available documents in the training set using say BERT topic modeling or else, and then generate a few QCA from each of the topics);

- Generate questions such that the <u>context is insufficient</u> to provide answers, so that the ground truth dataset also has enough ‘negative’ examples.

Retrieval Augmented Generation: How to identify thresholds …

Step-3: Determine the Threshold for the Metric and Cross-Validate

- E.g., compute Faithfulness for the risk of hallucination for the available QCA triplets in the ground truth data.

- Compute mean (say, μ_F and μ_{AR}) and standard deviation (say, σ_F and σ_{AR}) for each of them.

- Compute the confidence intervals. **Confidence Level** represents the degree of certainty that the AI system's performance will meet or exceed the threshold.

CI for Faithfulness: $\mu_F \pm$ (Z-score for given % confidence) σ_F
CI for Answer Relevance: $\mu_{AR} \pm$ (Z-score for given % confidence) σ_{AR}

E.g. (completely hypothetical),

- **High Risk Appetite**: **90% Confidence Level:** Willing to accept that in 10% of cases, performance may fall below the threshold.
- **Moderate Risk Appetite**: **95% Confidence Level:** Accepts only a 5% chance of performance falling below the threshold.
- **Low Risk Appetite**: **98-99% Confidence Level:** Aims for performance to meet thresholds in 98-99% of cases, allowing only 1-2% chance of falling below.

Retrieval Augmented Generation: How to identify thresholds …

Step-3: Compute the threshold for the given risk appetite

- E.g., for moderate risk appetite

Threshold Faithfulness = μ_F - (Z-score for 95% confidence) σ_F
Threshold for Answer Relevance = μ_{AR} - (Z-score for 95% confidence) σ_{AR}

Retrieval Augmented Generation: How to identify thresholds …

Even better… identify the threshold using cross-validations:

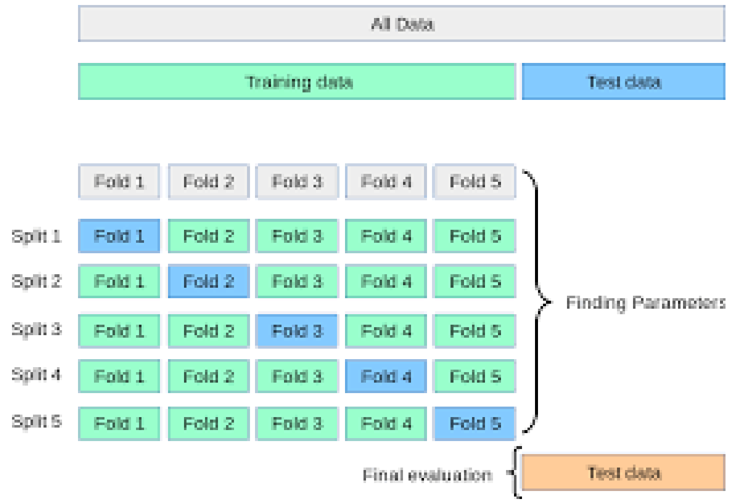
Step 1: Identify and quantify the relevant risks and risk appetite (e.g., 95% confidence level);

Step 2: Run the RAG system to generate outputs for the ground truth data, and calculate the evaluation scores.

Step 3: Take K-folds, and calculate statistical measures (mean, standard deviation) for K-1 folds, compute the threshold using $\mu - Z \times \sigma$.

Step 4: Check how the threshold did on the hold-out fold.

Step 5: Take an average of the thresholds (or the largest value) as the final threshold.



Even better… identify the threshold using cross-validations:

Step 1: Identify and quantify the relevant risks and risk appetite (e.g., 95% confidence level);

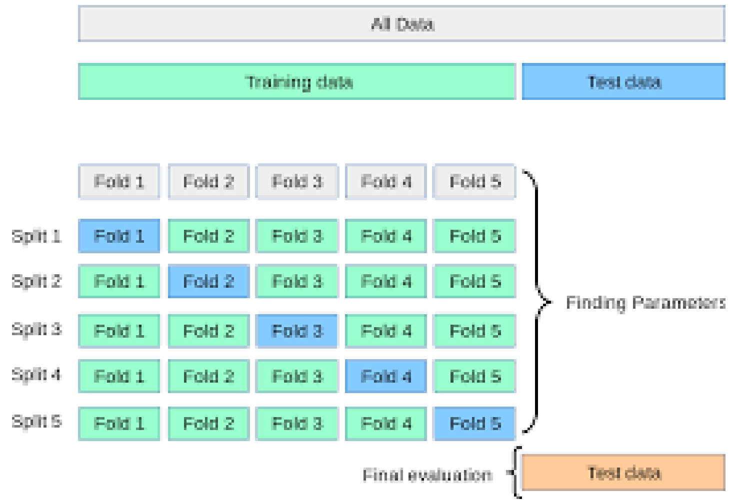
Step 2: Run the RAG system to generate outputs for the ground truth data, and calculate the evaluation scores.

Step 3: Take K-folds, and calculate statistical measures (mean, standard deviation) for K-1 folds, compute the threshold using $\mu - Z \times \sigma$.

Step 4: Check how the threshold did on the hold-out fold.

Even better, other statistical methods such as conformal prediction that does not assume normal distribution etc.

Step 5: Take an average of the thresholds (or the largest value) as the final threshold.



Other statistical methods for choosing thresholds

- Kernal Density Estimation to identify the ‘mid-point’ between the distributions of Pass and Fail labels;

- Compute AUC-ROC for a logistic regression between the faithfulness score as the input and the ground truth labels as the output, and pick the threshold as per the probability threshold;

- Instead of logistic regression, use a nonlinear model such as polynomial logistic regression or Generalized Additive Models (GAMs);

- Conformal prediction – distribution free, model agnostic choice methods to pick a threshold for a given confidence level;

- Etc.

Example Thresholds for Faithfulness on the HaluBench data

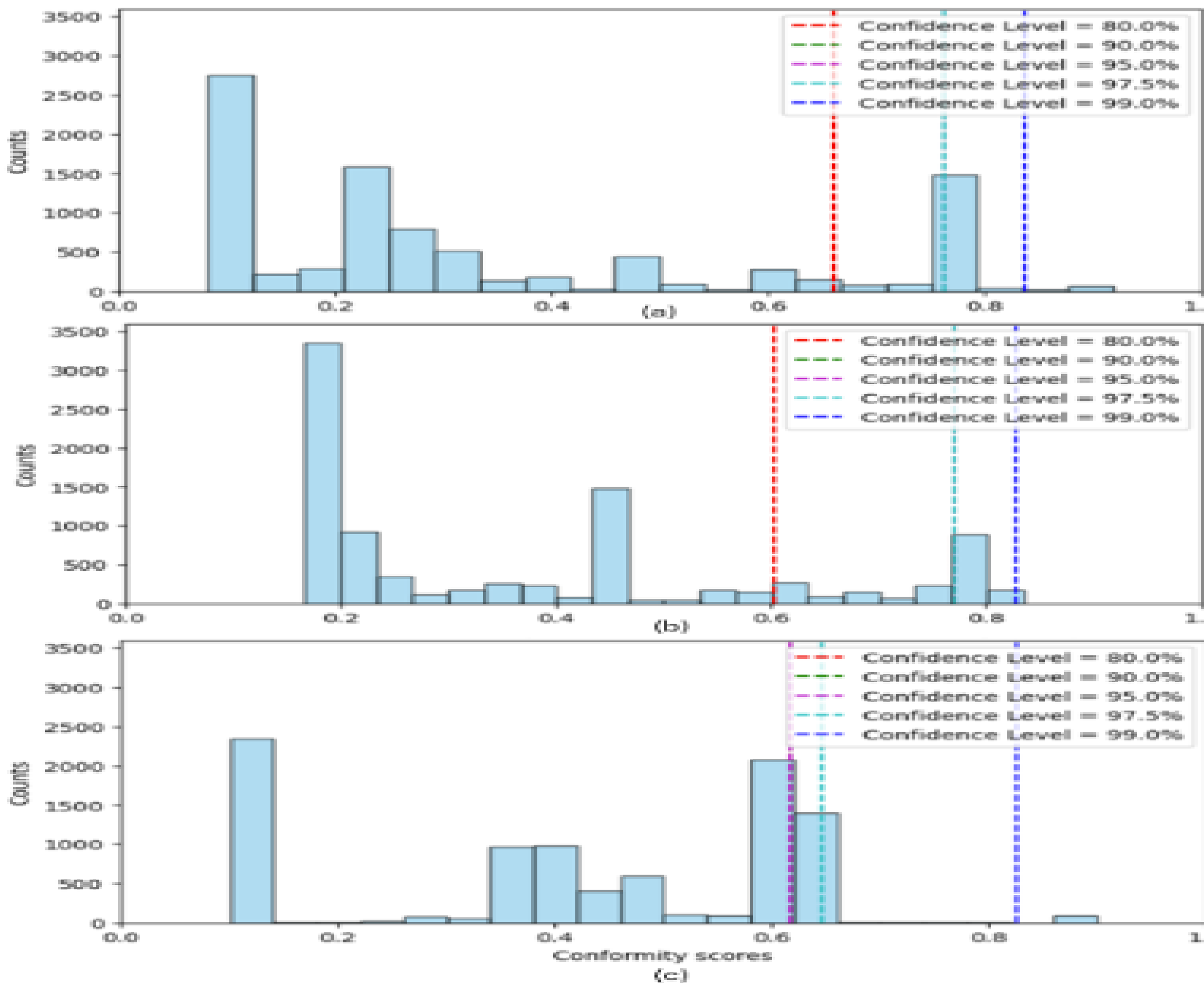
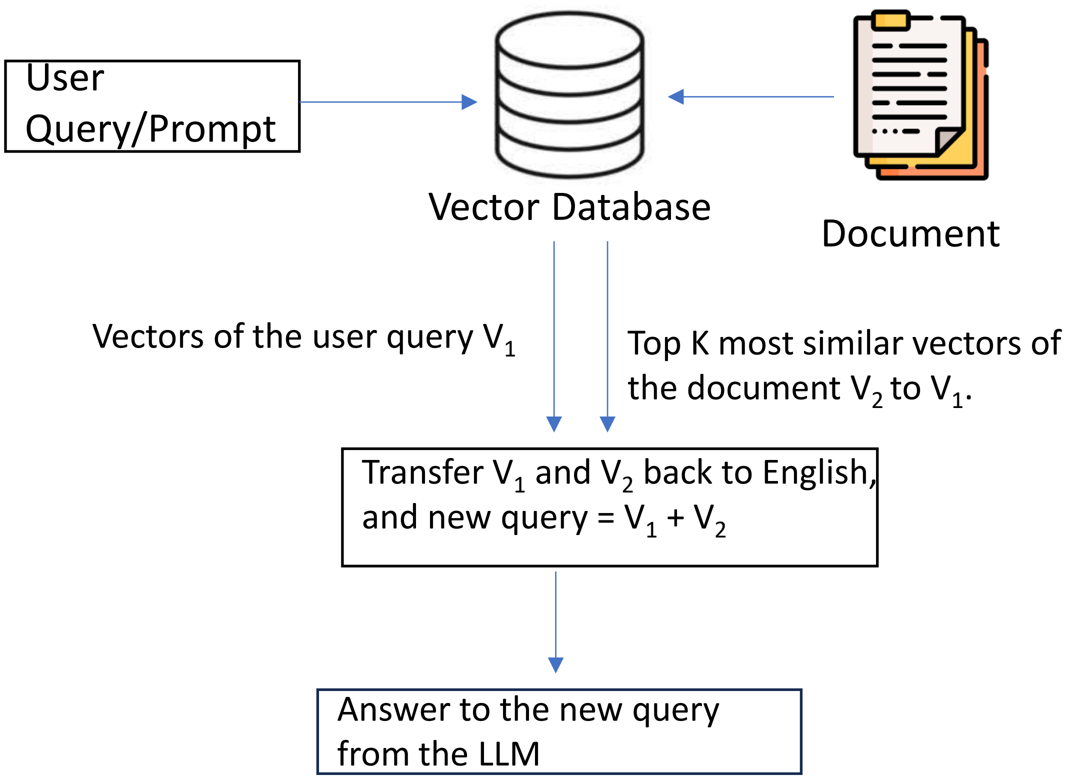


Figure 8: Distribution of conformity scores with thresholds. (a) UpTrain, (b) RAGAS, (c) DeepEval.

Conclusion

- Identifying threshold of an LLM evaluation metric is an immediate problem to be solved when developers need to deploy LLM applications;
- We provide a systematic process to determine the threshold based on risks of the application and risk appetite of the stakeholders;
- We also proposed various statistical methods to compute the threshold in practice;
- Applied these methods on a publicly available dataset for hallucination benchmark for the faithfulness metric.
- Future work: tackling multiple evaluation metrics and their threshold simultaneously.

An Application: Retrieval Augmented Generation



- Get embedding vectors for the user query from a language model (also called vector database).
- Get the embedding vectors for the concerned document(s) from the same vector database.
- PS: This language model can be any model, including TFIDF/BERT/GPT etc.
- Then, identify the K most similar vectors from the set of vectors of the documents.
- Now take convert both sets of vectors back to the language, i.e., 'augment' the original query with the 'context' text from the document, i.e.,
  new query = query + context.
- Get the answer from the LLM for 'new query'.